Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 1996 Proceedings

Americas Conference on Information Systems (AMCIS)

8-16-1996

Negotiating Requirements: A Social Cognitive Perspective on the Systems Development Process

Elizabeth J. Davidson

Detartment of Decision Sciences, College of Business Administration, University of Hawaii, davidson@busadm.cba.hawii.edu

Negotiating Requirements:
A Social Cognitive Perspective on the Systems Development Process

Elizabeth J. Davidson
Department of Decision Sciences
College of Business Administration
2404 Maile Way
University of Hawaii
Honolulu, HI 96822
davidson@busadm.cba.hawaii.edu

Introduction

Systems development methodologies typically model the systems development process as a series of steps in which definition of an information technology (IT) application is successively refined through design artifacts. Studies of software development in an organizational context suggest that this methodological approach does not adequately consider the organizational complexity of requirements definition (Bansler and Bødker 1993) nor the social aspects of the systems development process (Curtis, Krasner and Iscoe 1988). Instead, research suggests that the systems development process should also be addressed as a social process of learning, communication, negotiation, and conflict resolution (Bansler and Bødker 1993; Curtis, Krasner, and Iscoe 1988; Newman and Noble 1990; Newman and Robey 1992; Robey, Farrow, and Franz 1989; Walz, Elam, and Curtis 1993).

In this research, I adopted a social cognitive perspective (Orlikowski and Gash 1994) to investigate the negotiations of key stakeholders around IT requirements definition. In an in-depth, longitudinal field study of two systems development projects at one research site, I examined how project participants communicated knowledge, assumptions, and expectations as they attempted to negotiate a shared interpretation of IT requirements within and between groups. In this paper I present findings from this research to consider the influence and role of sensemaking devices and systems development artifacts in communication and negotiation processes and implications for systems development processes and outcomes.

Sensemaking devices in communication and negotiation processes

Systems development participants' use of sensemaking devices to communicate assumptions and expectations, articulate ideas, and share knowledge was pervasive in their face-to-face interactions. These sensemaking devices can be categorized as follows:

i) Narrative histories of the project;

ii) Organizational stories;

iii) Personal stories;

iv) Scenarios-of-use and vignettes;

v) Metaphors.

Sensemaking devices served as symbols for interrelated ideas, knowledge, assumptions and expectations. All types of systems development participants (technical developers, users, managers) used these symbols as a kind of short-hand notation in their communications and interactions. They drew on a shared understanding of the assumptions underlying the symbols as they tried to make sense of changes and events

in the systems development project. Because they were equivocal, sensemaking devices suggested multiple interpretations that could be applied in different situations. In these ways, sensemaking devices facilitated both communication among project participants and negotiation of common interpretations of the project and of IT requirements.

For example, one organizational story, commonly referred to as "the MIS Fiasco," related to a failed multi-million dollar project at the research site. The story, which had become an organizational symbol for unacceptable systems development approaches, provided organization members a quick way to express the assumption that large-scale, long term projects were a bad idea. Systems development participants who had no personal experience with that project drew on its story in a variety of contexts, for example, to interpret outcomes such as managers' rejection of their recommendations (*"After the MIS fiasco, no one wanted to commit to a major development style"*) and to decide how to phase delivery of IT requirements (e.g., *"If we bit off too big a piece, we'd have, pardon my analogy, an MIS Fiasco, Part 2, here.")* Mere mention of this powerful organizational story in a project team meeting would effectively bring participants into agreement on the need to curtail requirements and thus to limit the size and duration of their own project.

Use of sensemaking devices was pervasive in systems development participants' communications and negotiations, yet they used these devices tacitly, seldom stopping to reflect on the underlying assumptions and expectations embedded in their metaphors, stories, narratives, and so on. For example, while technical developers used the metaphor of a *data warehouse* extensively to communicate high-level images and concepts about the IT application, they did not examine the implications of the metaphor by asking each other specifically what this metaphor implied or contrasting this metaphor with other potential metaphors such as the notion of a data *department store* or *boutique*. By doing so, they might have surfaced critical differences in assumptions about the need for user support that instead arose during implementation of the system.

Systems development artifacts in communication and negotiation processes

Participants in the projects studied used four types of artifacts:

i) Analytic models (particularly data models);

ii) Information technology (e.g., existing systems; purchased software packages);

iii) Project work plans;

iv) System development and project documents.

These artifacts served three primary roles in communication and negotiation processes. First, team members drew on shared understandings of the assumptions about information technology and systems development underlying such artifacts. Second, they served as a focal point in interactions through which assumptions and expectations about the project and IT application were surfaced and explored and a shared interpretation of requirements was sometimes negotiated. Finally, they served as an outcome of negotiations which embedded participants' knowledge, interpretations, and decisions about requirements.

For example, by developing data models, technical developers tacitly accepted and shared assumptions such as the belief that identifying and describing data was the central task in defining requirements and that complex, even arbitrary business practices could be succinctly described in terms of relationships between data entities. As they focused on the data model in discussions, they spontaneously created scenarios-of-use, drew on metaphors, related incidents and stories that conveyed their knowledge of business processes and data use, and so on. In this way, they exchanged knowledge, assumptions and expectations about the development project, IT application, and requirements. The data model was an outcome of their

negotiations which embedded both discursive knowledge and tacit assumptions about information and data into descriptions of data and the data model.

Although systems development artifacts did facilitate technical developers' negotiations around requirements, particularly in their face-to-face interactions, these artifacts did little to facilitate communication or negotiation of a shared interpretation of requirements between technical developers and users. Users did not necessarily understand nor share assumptions underlying these artifacts. Since they did not participate with technical developers in the interactive creation of artifacts, they did not benefit from knowledge sharing that occurred in working sessions. As a result, artifacts primarily reflected developers, not users, assumptions and expectations about requirements. In fact, effective artifacts to facilitate communication and negotiation between technical developers and users during requirements definition activities were noticeably lacking. Only through demonstrations of the technology were technical developers able to communicate a surface-level understanding of the IT application to users. However, observation of IT features in demonstrations did little to surface critical differences between systems developers' and users' assumptions and expectations about implications of the IT application for work practices and job responsibilities. Instead, issues arose after initial implementation as users raised questions and objections to the new technology.

Implications for systems development processes and outcomes

Although all participants in systems development activities extensively and spontaneously used sensemaking devices in their interactions, sensemaking devices appeared infrequently, and then only tacitly, in systems development artifacts. The parsimonious structure of artifacts did not accommodate the unstructured narrative form of sensemaking devices used orally in face-to-face communications. Because knowledge exchange that occurred in face-to-face interactions through creation or application of sense-making devices was not preserved in or transmitted through artifacts, the value of both sensemaking devices and artifacts to facilitate sharing of knowledge, assumptions, and expectations *over time* and *among* individuals or groups was limited.

In particular, omitting sensemaking devices from systems development and project documents further limited the understanding of the project and of requirements that users gleaned from such artifacts. Without the sensemaking devices that animated communication and illustrated ideas in face-to-face interactions, these artifacts communicated little of substance to users. For example, although project history narratives were an important sensemaking device participants used in face-to-face interactions, system development and project documents provided only a point-in-time, sanitized version of project events in abstract terms of goals, objectives, tasks completed, and so on. Yet systems developers' interpretation of events in the project's history had a great influence on their decisions and recommendations about requirements contained in these artifacts. Similarly, "the MIS Fiasco" story did not appear in artifacts, although it had a powerful influence on systems developers' decisions about how and when to address IT requirements. Including these sensemaking devices in system and project documents could have provided users who review ed and approved documents with insight into developers' rationale for decisions and in this way enabled more meaningful negotiations between these groups concerning project direction, scope and IT requirements.

Conclusions

This research addresses social cognitive processes in systems development by examining how participants communicate and share assumptions and expectations to negotiate a shared interpretation of IT requirements. Developing and maintaining shared interpretations of requirements are critical in successful systems development efforts, and fostering a common understanding between technical developers and users can increase users' satisfaction with an IT application (Ginzberg 1981). This paper discusses the influence and role of sensemaking devices and systems development artifacts in these processes. Findings presented here suggest that both sensemaking devices and systems development artifacts play an important role in negotiations around requirements, and that sensemaking devices in particular are critical

communication devices in this social cognitive process. Other researchers have similarly noted the importance of sensemaking devices in systems development and problem-solving activities, for example, use of metaphors (Boland and Greenberg 1992; Mason 1991), stories (Brown and Duguid 1991), and scenarios-of-use (Walz, Elam, and Curtis 1993). Findings further suggest, however, that tacit, rather than explicit, use of sensemaking devices and their absence from systems development artifacts limits their value as mechanisms for communicating and sharing an understanding of requirements across time and between groups. Development of techniques and methods which actively employ sensemaking devices and which integrate them into design artifacts is needed to facilitate communication and negotiation of IT requirements, and in this way to improve systems development outcomes.

Acknowledgements

The author gratefully acknowledges the support of the Center for Information Systems Research at the Massachusetts Institute of Technology.

References

Bansler, J., and Bødker, K., "A Reappraisal of Structured Analysis: Design in an Organizational Context," *ACM Transactions on Information Systems* (11:2), 1993, pp. 165-193.

Boland, Jr., R., and Greenberg, R., "Method and Metaphor in Organizational Analysis," *Accounting, Management, and Information Technology* (2:2), 1992, pp. 117-141.

Brown, J., and Duguid, P., "Organizational Learning and Communities-of-Practice: Toward a Unified View of Working, Learning, and Innovation," *Organization Science* (2:1), 1991, pp. 40-57.

Curtis, B., Krasner, H., and Iscoe, N., "A Field Study of the Software Design Process for Large Systems," *Communications of the ACM* (31:11), 1988, pp. 1268-1287.

Ginzberg, M., "Early Diagnosis of MIS Implementation Failure: Promising Results and Unanswered Questions, *Management Science* (27:4), 1981, pp. 459-478.

Mason, R., "The Role of Metaphors in Strategic Information Systems Planning," *Journal of Management Information Systems* (9:2), 1991, pp. 11-30.

Newman, M. and Noble, F., "User Involvement as an Interaction Process: A Case Study," *Information Systems Research* (1:1), 1990, pp. 89-113.

Newman, M. and Robey, D., "A Social Process Model of User-Analyst Relationships," *MIS Quarterly* (16:2), 1992, pp. 249-266.

Orlikowski, W., and Gash, D., "Technology Frames: Making Sense of Information Technology in Organizations," *ACM Transactions on Information Systems* (12:2), 1994, pp. 174-207.

Robey, D., Farrow, D. and Franz, C., "Group Process and Conflict in Systems Development", *Management Science* (35:10), 1989, pp. 1172-1191.

Walz, D., Elam, J., and Curtis, B., "Inside a Software Design Team: Knowledge Acquisition, Sharing, and Integration," *Communications of the ACM* (36:10), 1993, pp. 63-77.